Parameter Page

Local Display Application

Sep 19, 1989

Introduction

The Parameter Page is a generic application that has been implemented on many platforms of the various accelerator control systems at Fermilab. It is designed to display a list of devices along with their current reading values (and other related values) scaled to engineering units and updated at a rate of about once per second. One can also control those devices which are controllable. A means is provided to select what devices make up the list of devices and to save such sets of devices. One can usually plot device data in real time at a rate of 15 Hz and observe the effect of knob adjustments of a device upon the data which is being plotted. These features are implemented in the Parameter Page application that runs on the Local Station display; in fact, the Parameter Page was the first applications to be written for the Local Station system and remains the mainstay.

Display layout

```
1 PREAMP PS       09/19/89 1404
TEMPT1 T1 TEMPERATURE  17.23  C
TEMPT2 T2 TEMPERATURE  16.71  C
TEMPT3 T3 TEMPERATURE  17.19  C
TEMPT4 T4 TEMPERATURE  16.61  C
TEMPT5 T5 TEMPERATURE  16.65  C
TEMPT6 T6 TEMPERATURE  16.74  C
TEMPT7 T7 TEMPERATURE  17.5   C
TEMPIN H20 TEMP IN     1.566  C
TMPOUT H20 TEMP OUT    19.14  C
DIDEOT DIODE TEMP OUT  16.93  C
DIDEIN DIODE TEMP IN   16.78  C
CTL+15 CONTROL +15V    15.15  V
CTL+5  CONTROL +5V     4.908  V
CTL-15 CONTROL -15V   -14.83  V
```

There are up to 14 display lines available for displaying device data. Information that can be displayed about one device is:

1. The 6-character device name
2. A minus sign used to indicate that the device is controllable
3. Up to 14 characters of the device title (skipping the first word)
4. Associated digital status bits (reduces title space)
5. Associated digital control bits interrupt buttons
6. Numeric value: reading, setting, nominal, tolerance, channel#, “set”
7. Alarm status: good/bad, beam inhibit
8. Engineering units text

Numeric field

The various modes of display are reading, setting, nominal, tolerance and “set” values, and they are selected in turn by the buttons labeled “`A/D`, `D/A`, `NOM`, `TOL`, and `SET`” on the small console. The first four options are obvious. The last one requires explanation. The name “`SET`” comes from its use as displaying the setting value of a controllable device saved when the line was last initialized—either upon entry to the page or upon manually entering that device on the line. If there has been no (net) change in the setting since that time, the node and channel# is displayed instead in the form “`NNNN:CCCC`,” where the node# `NNNN` and the channel# `CCCC` are displayed in hex. If a previous setting is displayed and one interrupts anywhere in the numeric field, the device will be restored to the saved value, and one should expect to see the saved value replaced by the node/channel representation. Note that the saved value that is restored is not read from the screen in order to prevent round-off problems.

The three unit selections available are engineering units, volts and hex, available for selection by the buttons labelled "ENG, VOLT, and HEX" on the small console. While engineering units or volts are selected, the values displayed are averages of the 15 Hz values, over 13 cycles, with preference given to cycles that have beam status asserted (see below). While hex is selected, the values displayed are merely sampled every 13 cycles. The reason for this is that hex may be used for checking for "stuck" or "missing" bits on an A/D reading, for example. The reason for normally displaying averages is simply to cut down on the noise. If volts or hex units is selected, the engineering units text field for each line is changed to read " V. " instead.

The alarm status is indicated by displaying the current value field in inverse video to indicate "bad" alarm status. If beam is inhibited because the inhibit option is selected for that device, the inverse video action is flashing.

Access to the alarm flag bits is not provided by the Parameter Page, except that a device can be observed to not be in the alarm scan (active flag bit=0) if the numeric field is blank when displaying nominal or tolerance values. Alarm flags for an analog channel can be accessed from the Analog Descriptor Page.

Engineering units scaling

For each analog channel there are 4 floating point constants that are used for engineering units conversion. The values are normally entered from the Analog Descriptor Page. The first two are used for scaling the reading, nominal and tolerance values; the third and fourth constants are used for scaling setting values. For each pair of constants, the first is a fullscale value, and the second is an offset, both in engineering units. All the scaling formulae are linear. To handle nonlinear cases, the raw data must be linearized first, perhaps by use of a Data Access Table entry that is designed for the purpose.

To scale a reading or a nominal value to engineering units, use the following formula:

```
eng:= Float(raw)/32768.*F1 + F2;
```

F1–F4 are the four floating point constants, and raw is the two-byte value.

To scale a tolerance to engineering units, use the following formula:

```
eng:= abs(Float(raw)/32768.*F1);
```

Note that the offset value is not used, as the tolerance is like a first derivative. Tolerances have no sign.

To scale a setting to engineering units, use the following formula:

```
eng:= Float(raw)/32768.*F3 + F4;
```

To derive the raw values from engineering units values, merely work the above linear formulae backwards.

A special case exists for motors. When entering a setting for a motor, one is really entering a desired reading, so the first two constants get used, rather than the last two. The system software actually uses the third constant in this case as the number of motor steps required to make a 10 volt change in the reading, to compute the number of steps to issue to the motor.

Beam status

Beam cycles are indicated by status bit `0x009F` in the local station having the value zero. The averaging logic attempts to select those cycles which have beam status asserted for inclusion in the averaging accumulation. In the event that no cycles have beam status, then the averages include all those non-beam cycles. If all cycles have beam status, then the averages include all those beam cycles. This means that if the status bit never changes, the averages simply include the data from all the cycles. It's when the beam status varies that it gets interesting. If the average has started with a non-beam cycle, but a beam cycle occurs later in the series of 13 cycles that make up the average set, then the average is started over with that beam cycle, and only data from beam cycles will be included over the next 12 cycles. A result of this is that a beam cycle occurring at 1 Hz, say, will result in numeric values being updated with that beam cycle's values at 1 Hz. Or, a burst of 5 beam cycles repeating every 2 seconds will result in averages of the 5 beam cycles alternating with averages of 13 non-beam cycles. The updating attempts to track beam cycles.

Analog control

Analog devices can be controlled by typing a value into the numeric field and pressing the keyboard interrupt button, with the cursor positioned immediately after the last character typed. (This is referred to as entering a value and "interrupting." The significance of the term "keyboard interrupt" is steeped in history.) In this way, setting, nominal, or tolerance values can be entered in the currently selected units. The updating of a device's current value is inhibited while the cursor remains within the numeric field in order to prevent overwriting the value being typed.

A special form of input is permitted for stepping motor control. A number followed by an "`S`" within the numeric field is interpreted as the number of steps to issue to the motor. Without the "`S`" included, the value entered is accepted as a desired reading in the current units. The system uses the 3rd floating point constant in the analog descriptor, which for a motor-controlled device is the number of steps to make a 10 volt change in the reading, in order to decide how many steps to issue to change the reading from its current reading value to the desired value. The process is not iterated. A subsequent adjustment of a motor which is moving will override its current control. One can therefore stop a motor that is moving to some far-ranging destination by either typing a "`0S`" or by adjusting the knob slightly.

Another means of analog control is through the use of the knob on the console, with the cursor positioned anywhere on the line of the device to be adjusted. The knob adjusts only the setting no matter what value is displayed. The settings are issued and the numeric value is updated at 15 Hz (with instantaneous values, not averages) while the knob is being turned so that one can more precisely follow the adjustments as they are being made. The update period for numeric values on lines whose devices are not being adjusted is about 0.8 seconds (13 cycles @ 15 Hz). An alternative to using the knob for analog control is the use of the raise/lower buttons on the console. They act to control the device at a rate of 16 "knob clicks" every 15 Hz cycle, which is about one turn of the knob per second. Knob control is related to the adjustment of the setting of a device through automatic scaling that depends only upon the analog control type that is used by the given device. An internal table in the system software (in the `SETAC` module) relates the knob sensitivity to the analog control type`#`. For control of stepping motors, one knob click corresponds to one motor step.

Knob adjustments are made in the station that is interfaced to the device. That node always knows the most recent setting value, as all setting commands to that device must pass through that station. The knob adjustment is scaled and added to the most recent setting to produce the new setting. If the allowed ±10 volt range is exceeded, the setting value is clamped to the appropriate end of the range. Some analog control types, such as timers, may be clamped to zero at the low end.

One other special form of analog control is applied to the nominal value of a device. An interrupt in the engineering units text field copies the present reading of the device to the nominal value. If one has just tuned a device to a new setting, this interrupt updates the nominal value to match.

Digital status

Associated digital status bits can be displayed on the same line as the analog value. If there is one associated digital status bit, the last 6 characters of the title field are replaced by two 3-character fields to indicate which of the two states of status is current, such as " `ON...`" or "`...OFF`", for example. (The periods here indicate that the two states represent control buttons described in the next section. If no control is available related to the status, then spaces are shown in place of the periods.)

For two associated status bits, the first one (as described in the analog descriptor digital control field) is displayed as just described. The second one is displayed in another 6-character field just to the left in a similar manner, and no title is displayed as there is no room for it (either on the display or in the descriptor title field).

Digital control

Digital control buttons can be designed to work in conjunction with the corresponding digital status. Interrupt under the 3-character field that is represented by periods to execute the control that would presumably result in the desired change of status. In the above example, with the status field displayed as "`...OFF`", for example, interrupt under the periods to make the device turn `ON`. It is also possible to have digital control without a corresponding display of status, such as might be used for a " `RESET`" control. In this case, the field would always display the same way, and an interrupt anywhere in the field would issue the `RESET` control to the device. Details on how to set up the analog descriptor to accomplish the many and various types of digital control that are supported by the system are covered in a separate document entitled "*Digital Control Pulse Delays*".

Error status

An error status code number is displayed in the second character position of the top (title) line. If there are no errors, this character will be blank; otherwise, its value is displayed in inverse video. Likely values that may be displayed are:

8 At least one node is not responding to the request for data.

7 The requested data seems to be tardy, but data has been received at least once from all nodes. This is common if nodes are not synchronized.

Device entry

There are two principal means of entering a device to be displayed on a line. One is by entering the device name (from 2–6 characters) at the beginning of the line, over any name already displayed there, and interrupting. If the name is not the name of a local device, the system uses the network to find another station that has a device by that name. (This is done by a broadcast message to the network and results in almost no perceptible delay.) If there is no response from the network, a blank line results.

The other means of device entry is by entering the `node:chan` representation and interrupting. If the device with that `node:chan` does not exist or if the node is unreachable via the network, then a generic form of parameter line results that includes the `node:chan` designation. If the node is down and subsequently comes back up, the line will automatically re-initialize with the proper data showing. This happens because the system automatically reissues active requests to silent nodes every 2 seconds.

All such changes made to the list of displayed devices are temporary unless they are "saved." To make the current list of channels the "permanent" list—the one displayed when the page is subsequently re-invoked—type an "`X`" in the upper left corner of the display (use the Home key) and interrupt. The current list will be saved and the display refreshed.

More details of variations of device entry are found in the document entitled *"Entering New Parameters on a Local Station"*.

Plotting

A somewhat crude form of plotting is supported using the "semi-graphics" mode of the display chip used on the Crate Utility board. This mode allows display of a pattern of 6 square dots (2 wide by 3 high) at each character position on the screen. For the Parameter Page support, an 11-line portion of the display is used for the graphics plot using this mode. This gives a resolution of 64 dots of width and 33 dots of height. The remaining three lines (out of the 14 normally used for parameters) are used for entering scale ranges and two parameter lines. The two parameter lines are used to select the channels to be plotted and are also available for control in the usual way.

There are two forms of plotting control supported. One is a time plot—plotting one channel's reading against time—and the other is a parameter plot—plotting two parameters against each other.

To enter the plotting mode, place the cursor at the start of the 12th parameter line. (Pressing the Home key and 4 up-arrows will do this.) Type a "`T`" or a "`P`" and interrupt to enter the time plot or the parameter plot mode respectively.

Upon entering the time plot mode, the 12th line will read as follows:

```
YI,F,TIME=0,10,60
```

or something like that. The three values shown are the initial y-value, which will be plotted at the line above; the final y-value, which will be plotted at the top of the line after the page title; and the number of seconds to cross the screen from the left edge to the right edge. The three values can be altered by typing in free form with the cursor position respected. Whenever new values are entered and the interrupt is given, the plotting area is blanked and data from the y-channel (displayed on the next line) is plotted at 15 Hz. When the x-position reaches the right edge of the screen, the data continues to plot, although the x-position no longer changes. To restart the plot with the same scaling, interrupt to the right of the values shown. To continue the plot at the left edge without erasing what is already plotted, interrupt with the cursor to the left of the equal sign (but not at the start of the line). As the plotting of data points progresses, an attempt is made to show the most recent data value plotted as a bright dot and the old data as a dim(mer) dot. The word "attempt" here refers to the fact that the bright/dim option in the semi-graphics mode can only be applied on a character position basis (2 dots by 3 dots).

Upon entering the parameter plot mode, the 12th line will read as follows:

```
            YI,F,XI,F=0,10,0,10
```

or something like it. Here the scaling values are the initial and final y-values followed by the initial and final x-values. The y-parameter is the device displayed on the next line, and the x-parameter is the device displayed on the following (last) parameter line. In this mode the current x-data and y-data values are plotted continuously at 15 Hz. The knob can be used to adjust, say, the x-parameter to observe the effect of its changes on the y-parameter. If the selected display mode is settings, then the x-data value (but not the y-data) plotted is the setting value, rather than the reading value. Other variations are similar to the time plot.

In either mode, device data can be plotted in either engineering units or in volts. To exit the plotting mode, interrupt at the start of the 12th line. The lines of devices that were preempted for the plotting area will reappear. One can make the plotting mode "permanent" by typing an "`X`" in the corner as described above for saving the current list of channels.

Internal matters

The use of the `conv` byte in the analog descriptor has changed. It used to have values of 1–4 to signify (`normal`, `rfGrad`, `rfPower`, `zeroData`) attributes of a channel. It is now coded in a bit-significant way, where bit #1 (mask=`0x02`) means `captureData` and bit #2 (mask=`0x04`) means `zeroData`. In this way, a channel can have both attributes using a value of `0x06`. The Parameter Page was written prior to this change. There may be adjustments that should be made in the light of the new `conv` usage.

Many details not discussed here can be found in the source code, which consists of about 1100 lines of Pascal compiled into about 10K bytes of code.